

Name:

ID:

Section: serial:

[18 pts]

QUESTION TWO:

- o Give ONE statement that defines an array BB consisting of 200 signed word values.
BB 200 dup (?)
- o Give ONE instruction that divides the value in register BX by 2.
SHR BX, 1
- o Give ONE instruction that inverts ONLY all odd-numbered bits in DX register.
DBT DX, 00000001H
- o Give ONE instruction that swaps the two halves of register ECX.
RCL ECX, 16
- o Give ONE instruction that copies the value stored on the top of the stack into the memory location named delta.
POP delta
- o Give ONE instruction that causes the IP/EIP contents to be pushed onto the stack.
PUSH EIP
- o Give ONE instruction that causes the stack pointer register to be incremented by 4 (Add not allowed).
ADD SP, 4
- o Give ONE instruction that stores in EDX register the signed value stored in CL register.
MOVSDI ECX, CL
- o Give ONE instruction that invokes a NEAR procedure offset address its first instruction is stored in BX.
JMP BX
- o Give ONE instruction that makes the carry flag equal to the bit number 10 in CX register.
STC
- o Consider the following register dump in a system operating in real-address mode and answer questions a to d [All values in Hexadecimal]

CS=6FA0	DS=7FA0	SS=8FA0	SP=2A00	BP=2CA8	IP=20F8
AX=2A5B	BX=1AF5	CX=64FC	DX=0CDA	SI=35CF	DI=3500

- a) The physical address of the next instruction is 71A08
- b) The logical address of the next instruction is 71A08
- c) The type of the source operand used in this instruction: ADC AL, U[BX] is register
- d) If the instruction: PUSH CX is executed, the SP will contain 29FC H

- o A1 SBYTE -128, 255
- o B1 SWORD 128, -1, -255
- o MT WORD 20 DUP (30, 90)
- o C1 equ "A"
- o F1 equ

- 1) The instruction MOV AX, B1+4 stores the value FF01 H in AX register.
- 2) The instruction MOVSBX BX, A1 stores the value 5F8 H in BX register.
- 3) The instruction MOV DL, byte ptr A1+2 stores the value 10 H in DL register.
- 4) To calculate the number of bytes used by all above statements, the blank in the last statement must contain 85

Name: _____
QUESTION THREE:

id: _____

section: _____ serial: _____
[18 pts]

- o Write No more than 3 instructions to subtract $M3 = M3 - 2 * M4$, where M3 and M4 are predefined signed memory words.

`mov ebx, M3`
`mov ecx, M4`
`sub ebx, ecx`

- o Assuming AL and BL contain any unsigned values, Write NO more than 5 instructions to compute: $ECX = BL * 128 - AL * 32$. Using MULTIPLY instructions is NOT allowed.

`mov ecx, 128`
`mov ebx, BL`
`mul ebx`
`mov ecx, 32`
`mov ebx, AL`
`mul ebx`
`sub ecx, ebx`

- o Write NO MORE THAN 4 instructions to subtract $M4 = M4 - M3$, where M3 and M4 are quadword memory locations (8 bytes each) defined as shown below:

M3 Qword 70F05060D0801020H
M4 Qword 6020E03090C010F0H

`mov ecx, M3`
`sub M4, ecx`

- o Assuming: X and Y are predefined signed bytes, Write the needed instructions to compute: $EBX = (2^8 - X) * (Y + 7)$.

`mov ecx, 256`
`sub ecx, X`
`mov ebx, Y`
`add ebx, 7`
`mul ecx, ebx`

Name:

id:

section: serial:

QUESTION FOUR:

[20 pts]

- 1) The assembler accepts as input a _____ program and produced as output an _____ program.
a) .obj, .asm b) .asm, .obj c) .asm, .exe d) .obj, .lst e) None
- 2) The directive "M1 dword 32ABFFH" is equivalent to:
a) M1 WORD 32ABH, 0FFH b) M1 WORD 0ABFFh, 32H
c) M1 WORD 0FFABh, 32H d) M1 WORD 0AB32, 0FFH e) None
- 3) The instruction that makes EBX point to the element FF[20] in a signed word array named FF is:
a) MOV EBX, FF[40] b) MOV EBX, OFFSET FF[20]
c) LEA EBX, FF[20] d) LEA EBX, FF[40] e) None
- 4) Give 1 assembly statement to define a constant, called MONTH, having a value 24*30.
a) month equ 24*30 b) month byte 24*30 c) month sbyte 24*30
d) month word 24*30 e) None
- 5) The directive "TT dword 2 dup(20ACH, 4 dup(3, 9950), 77)" occupies _____ bytes:
a) 80 b) 40 c) 80H d) 32 e) None
- 6) If ax = C971H, the execution of instruction "ROR AX, 8" stores in AX:
a) 179C b) 17C9 c) 71C9 d) 719C e) None
- 7) If register ax contains A4CE, then after executing SAR ax, 4 register ax will contain:
a) 0A4C b) FA4C c) A4CF d) 4CEA e) None
- 8) The range of signed numbers that can be stored in ONE BYTE is from _____ H to _____ H.
a) 0 - 255 b) -128 - +128 c) -128 - +127 d) -255 - +255 e) None
- 9) The value (11001010)₂ represents a signed number (_____) ₁₀, and unsigned number (_____) ₁₀.
a) 202, -54 b) -74, 202 c) -202, 54 d) -54, 202 e) None
- 10) A computer has 24 address lines and 20 data lines. The maximum memory size directly addressable by this computer is _____ bytes.
a) 2²⁰ b) 2²⁴ c) 2²⁴ / 1024 d) 2²⁴ - 1 e) None
- 11) If a computer has 128 Mbytes main memory, the minimum number of address lines needed for this computer is _____.
a) 128 b) 64 c) 32 d) 25 e) None
- 12) The proc statement that defines a procedure named P9 with 2 parameters: a1 of type word, b1 is a pointer to byte is:
a) P9 proc, a1: word, b1: byte b) P9 proc a1: word, b1: byte
c) P9 proc a1: word, b1: byte d) P9 proc, a1: word, b1: ptr byte
e) None
- 13) The statement that calls a procedure P9 defined in the previous question with arguments: bx and a predefined byte named KK is:
a) invoke p9, ax, KK b) call p9, ax, KK
c) invoke p9, ax, addr KK d) invoke p9, ax, *KK e) None
- 14) The address of the next instruction is specified by _____: _____ registers.
a) cs:ip b) cx:ip c) cx:eip d) cs:sp e) None

Name:

id:

section: serial:

- 15) The range of unsigned numbers that can be stored in ONE word is from _____ H to _____ H
 a) $0 - 2^{16}$ b) $-2^{16} - (2^{16} - 1)$ c) $-2^{16} - 2^{16}$ d) $0 - (2^{16} - 1)$ e) None
- 16) _____ H: 720h = 12050h.
 a) 1193 b) 11930 c) 12770 d) Unknown e) None
- 17) Assembly program contains 2 types of statements: directives and _____.
 a) Instructions b) Segments c) Procedures d) b+c e) None
- 18) In real mode, if the address of the last byte in a data segment is 4FFCF, then the DS register will contain _____ H.
 a) 3FFD0 b) 4FFC c) 4FFC0 d) 3FFD e) None
- 19) The debug command that displays the source instructions starting at offset 6780H and ending at offset 68FFH is:
 a) D 6780 68FF b) A 6780 68FF c) U 6780 68FF
 d) A 6780H 68FFH e) None
- 20) The debug command that displays the next 32 bytes starting at offset 3F00 in the code segment.
 a) D CS:32 b) D CS:3F00 32 c) D CS:3F00 3FFF
 d) D CS:3F00 3F1F e) None

QUESTION#	1	2	3	4	5	6	7	8	9	10
ANSWER										

QUESTION#	11	12	13	14	15	16	17	18	19	20
ANSWER										

re:

id:

section:

serial:

STION FIVE: Write assembly instructions equivalent to the following C++ code

[8 pts]

```
i, j=0, k=0;
b[50],c[50],a[50]={10, 34, ...};
(i=0; i<50; i++)
if(*(a+i)%2 == 0)
{ *(b+k) = *(a+i); k++; }
else
{ *(c+j) = *(a+i); j++; }
cout << j << " " << k << endl;
```

WWW.UOBVIEW.COM

Uploaded By Smart ErrorCall

eax, write, duc
eax, write, duc
eax, write, duc
eax, write, duc

